



**TUGAS AKHIR - TE 145561**

**RANCANG BANGUN PERANGKAT KERAS DAN  
PENGATURAN DERAJAT KEBEBASAN PADA  
PENEMBAK PORTABEL**

Galih Rizki Wahyudhi  
NRP 2214030072

Dosen Pembimbing  
Suwito,ST.,MT

PROGRAM STUDI KOMPUTER KONTROL  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----



**FINAL PROJECT - TE 145561**

***DESIGN OF THE HARDWARE DEVICE AND  
SETTING DEGREE OF FREEDOM IN A PORTABLE  
SHEET***

Galih Rizki Wahyudhi  
NRP 2214030072

Advisor  
Suwito,ST.,MT

***COMPUTER CONTROL STUDY PROGRAM  
Electrical and Automation Engineering Department  
Vocational Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017***

-----Halaman ini sengaja dikosongkan-----

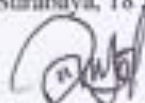
## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **"RANCANG BANGUN PERANGKAT KERAS DAN PENGATURAN DERAJAT KEBEBASAN PADA PENEMBAK PORTABEL,"** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 18 Juli 2017



Galih Rizki Wahyudhi  
NRP 2214030072

-----Halaman ini sengaja dikosongkan-----

**RANCANG BANGUN PERANGKAT KERAS DAN  
PENGATURAN DERAJAT KEBEBASAN PADA PENEMBAK  
PORTABEL**

**TUGAS AKHIR**

Disajikan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada

Program Komputer Kontrol  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing



Suwito ST, MT

NIP. 19810105 200501 1 004

**SURABAYA  
JULI, 2017**

-----Halaman ini sengaja dikosongkan-----



# **RANCANG BANGUN PERANGKAT KERAS DAN PENGATURAN DERAJAT KEBEBASAN PADA PENEMBAK PORTABEL**

**Nama : Galih Rizki Wahyudhi**  
**Pembimbing : Suwito,ST.,MT**

## **ABSTRAK**

Untuk mengatasi terjadinya peperangan, maka pada setiap negara harus memiliki suatu sistem pertahanan yang dimana mempertahankan sebuah negara ketika terjadi peperangan. Tetapi dalam mempertahankan sebuah negara tidaklah mudah, sehingga jatuhnya korban dalam peperangan tidak mungkin bisa dihindari. Akibat dari banyak jatuhnya korban menjadikan manusia sebagai pertahanan negara merupakan hal yang tidak efisien dan untuk meningkatkan kekuatan militer dengan pengendalian senjata jarak jauh yang sekarang ini masih kurang di dalam negeri. Untuk itu alat ini menggunakan Android sebagai pengontrolnya yang disambungkan dengan Wi-Fi pada minimum sistem sebagai telekontrol yang nantinya akan dikendalikan Android dan terdapat trackbar sebagai pengendali alat dan juga terdapat *interface* dari kamera yang ditempatkan pada alat. Alat ini menggunakan motor DC sebagai penggeraknya yang dimana terdapat 3 motor DC didalamnya yang masing-masing berguna sebagai pengatur sudut *vertical* dan *horizontal* serta sebagai penarik atau pengaktif tembak. Hasil yang di harapkan yaitu tingkat kepresisian dalam menembak sasaran dan jarak dari kontrol pengirim data dengan wifi yang memiliki jarak diatas 15 meter.

**Kata Kunci :** *minimum sistem, motor DC.*

-----Halaman ini sengaja dikosongkan-----

## ***DESIGN OF THE HARDWARE DEVICE AND SETTING DEGREE OF FREEDOM IN A PORTABLE SHEET***

***Name : Galih Rizki Wahyudhi***

***Advisor : Suwito,ST.,MT***

### ***ABSTRACT***

*To overcome the occurrence of war, then in each country must have a defense system in which to defend a country when the war occurred. But in defending a country is not easy, so the casualties in warfare can not be avoided. The consequences of many casualties making humanity a state defense are inefficient and to increase the military power with remote control of weapons that are currently lacking in the country. For that tool uses Android as a controller connected to Wi-Fi on a minimum system as telekontrol which will be controlled Android and there is a trackbar as a tool controller and there is also an interface of the camera placed on the tool. This tool uses a DC motor as a driving force where there are 3 DC motors inside which are respectively useful as vertical and horizontal angle adjusters and as towing or firing. The expected result is the high precision in shooting the target and the distance from the data sender control with wifi that has a distance above 15 meters.*

***Keywords : minimum sistem,DC motor***

-----Halaman ini sengaja dikosongkan-----

## KATA PENGANTAR

Puji syukur penulis junjatkan kehadirat Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Bidang Studi Komputer Kontrol, Jurusan D3 Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

### **RANCANG BANGUN PERANGKAT KERAS DAN PENGATURAN DERAJAT KEBEBASAN PADA PENEMBAK PORTABEL**

Kami mengucapkan terimakasih kepada semua pihak yang telah mendukung dalam pembuatan buku tugas akhir ini, termasuk dosen pembimbing kami, yaitu Suwito.ST,MT. yang telah menyempatkan waktunya untuk membimbing kami dalam menyelesaikan tugas akhir ini. Dan tidak lupa terimakasih kepada kedua orangtua atas dorongan semangat dan do'a dari mereka yang memudahkan kami dalam menyelesaikan buku tugas akhir.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 18 Juli 2017



Penulis

-----Halaman ini sengaja dikosongkan-----

# DAFTAR ISI

	HALAMAN
HALAMAN JUDUL.....	i
HALAMAN JUDUL.....	iii
PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
HALAMAN PENGESAHAN .....	vii
ABSTRAK .....	ix
<i>ABSTRACT</i> .....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xix
 BAB I PENDAHULUAN .....	 1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	1
1.3 Batasan Masalah.....	2
1.4 Tujuan .....	2
1.5 Metodologi Penelitian .....	2
1.6 Sistematika Laporan.....	2
1.7 Relevansi .....	3
 BAB II TEORI DASAR .....	 5
2.1 Arduino Uno .....	5
2.1.1 Ringkasan.....	6
2.1.2 Daya .....	7
2.1.3 Memori.....	7
2.1.4 <i>Input</i> dan <i>Output</i> .....	7
2.1.5 Komunikasi .....	8
2.2 Modul Wifi ESP8266.....	9
2.3 Motor DC .....	10
2.4 Modul L298N.....	11
 BAB III PERANCANGAN DAN PEMBUATAN ALAT .....	 13
3.1 Pembuatan Perangkat Elektronik .....	14
3.1.1 Skematik <i>Hardware</i> Kontrol .....	14

3.1.2	Pemilihan Jenis Motor DC .....	15
3.1.3	Pengkabelan <i>Axis</i> Penggerak Motor DC dan L298N .....	16
3.1.4	Pengkabelan Minimum Sistem dengan Sensor Resistif Sebagai Pengukur Sudut. ....	16
3.1.5	Pengkabelan pada Pematik Tembak.....	17
3.2	Pembuatan Perangkat Mekanik.....	17
3.2.1	Pembuatan Dudukan Atas .....	17
3.2.2	Dudukan bagian bawah .....	20
3.3	Pembuatan Perangkat Lunak.....	22
3.3.1	Pembuatan <i>Flowchart</i> Program.....	22
3.3.2	Pendefinisian Pin dan Pustaka .....	25
3.3.3	Segmen Program untuk Pengaturan Awal .....	26
3.3.4	Segmen Program untuk Menjalankan motor DC .....	27
BAB IV PENGUKURAN DAN PENGUJIAN .....		29
4.1	Pengukuran dan Pengujian Arduino dengan L298N dan Motor DC.....	29
4.2	Pengujian Respon Kontrol Motor DC.....	32
4.3	Pengujian Respon Penerimaan Data Ke Motor DC secara Berurutan .....	33
BAB V PENUTUP .....		35
5.1	Kesimpulan .....	35
5.2	Saran .....	35
DAFTAR PUSTAKA .....		37
LAMPIRAN .....		39
A.	<i>Listing</i> Program.....	39
B.	<i>Flowchart</i> .....	43
C.	<i>Datasheet / Specification</i> .....	45
C.1.	L298N .....	45
C.2.	Arduino UNO .....	57
RIWAYAT PENULIS .....		61



## DAFTAR GAMBAR

	HALAMAN
Gambar 2.1 Arduino UNO .....	6
Gambar 2.2 Modul L298N .....	11
Gambar 3.1 Diagram Blok Fungsi Cara Kerja Alat .....	13
Gambar 3.2 Blok Diagram Kontrol .....	14
Gambar 3.3 Skematik <i>Hardware</i> Kontrol .....	14
Gambar 3.4 <i>Gearbox</i> Menyilang .....	15
Gambar 3.5 <i>Gearbox</i> sejajar .....	16
Gambar 3.6 Pengkabelan <i>Axis</i> Penggerak Motor DC dan L298N ....	16
Gambar 3.7 Pengkabelan Sistem Arduino Uno dengan Sensor Resistif .....	17
Gambar 3.8 Pengkabelan pada Pematik Tembak .....	17
Gambar 3.9 Poros Inti Putar .....	18
Gambar 3.10 Penjepit Tembak .....	19
Gambar 3.11 Dudukan Atas Tembak .....	19
Gambar 3.12 kaki penyangga .....	20
Gambar 3.13 Bagian Penyangga Atas dan Bawah .....	20
Gambar 3.14 Bearing Tampak Bawah .....	20
Gambar 3.15 Bearing tampak Atas .....	21
Gambar 3.16 Bearing Tampak Samping .....	21
Gambar 3.17 Penggabungan Penyangga Bawah .....	21
Gambar 3.18 <i>Flowchart</i> Gerak Motor 1 .....	23
Gambar 3.19 <i>Flowchart</i> Gerak Motor 2 .....	24
Gambar 3.20 <i>Flowchart</i> Pelatuk Tembak .....	25
Gambar 3.21 Program Definisi Pustaka dan Tipe Data .....	25
Gambar 3.22 Segmen Program Untuk Pengaturan Awal .....	26
Gambar 3.23 Segmen Program Untuk Menjalankan Motor DC .....	27
Gambar 4.1 Gambar Keseluruhan Alat .....	29
Gambar 4.2 Gambar Grafik Gerakan Vertikal dan Horizontal Pada Tegangan 12 Volt .....	31
Gambar 4.3 Gambar Grafik Gerakan Vertikal dan Horizontal pada Tegangan 15 Volt .....	32
Gambar 4.4 Gambar Grafik Gerakan Vertikal dan Horizontal pada Tegangan 17 Volt .....	32

-----Halaman ini sengaja dikosongkan-----

## DAFTAR TABEL

### HALAMAN

Tabel 4.1 Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 12 Volt .....	30
Tabel 4.2 Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 15 Volt .....	30
Tabel 4.3 Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 17 Volt .....	31
Tabel 4.4 Respon Kontrol Motor DC .....	32
Tabel 4.5 Respon Motor DC Diberi Input Secara Berurutan dalam jarak 6 meter .....	33
Tabel 4.6 Respon Motor DC Diberi Input Secara Berurutan dalam jarak 15 meter .....	34

-----Halaman ini sengaja dikosongkan-----

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi banyak mempengaruhi perkembangannya peradaban dan kebutuhan manusia. Tidak terkecuali di bidang pertahanan, keamanan, militer, informasi. Kemajuan ilmu pengetahuan dan teknologi dalam bidang pertahanan dapat memberikan ancaman tersendiri, baik itu militer maupun non militer. Untuk itu, kemajuan Iptek harus dimanfaatkan untuk mendukung terwujudnya pertahanan negara yang solid, kokoh, dan kuat. Seiring derasnya arus globalisasi yang mempengaruhi segala aspek kehidupan berbangsa dan untuk mendukung pertahanan negaranya.

Dengan Iptek, sistem persenjataan dan alat peralatan baru dapat diciptakan untuk mendukung keperluan militer/pertahanan yang lebih handal, lebih akurat, dan lebih cepat dan fleksibel pengerahannya. Teknologi dalam memproduksi persenjataan dan alat peralatan tersebut terus berkembang sejalan dengan perkembangan Iptek. Teknologi pada bidang kemiliteran sangat penting diperlukan untuk pertahanan dan keamanan negara. Sekarang ini penggunaan robot sebagai pengganti peran manusia telah banyak diterapkan di berbagai bidang. Utamanya bidang yang sangat besar resikonya, sebagai contoh dalam pengintaian dan penembakan musuh yang dianggap sangat penting pada sistem keamanan militer pada bidang pertahanan dan keamanan.

Untuk itulah diperlukan sebuah sistem keamanan yang mampu memberikan serangan. Dengan harapan, sistem keamanan dengan pertahanan tipe seperti ini akan menimbulkan tekanan dan kesan lebih sulit dan lebih berbahaya bagi musuh. Dalam proyek tugas akhir ini merancang telekontrol penembak *portabel* menggunakan *micro controller* dan Android sebagai pengontrolnya. Alat ini terhubung menggunakan Wi-fi sebagai komunikasi data untuk pengiriman sinyal kontrol dari Android ke *micro controller* untuk di jalankan sesuai dengan yang di perintah dari Android. Fitur yang ada pada Android yaitu terdapat 2 tombol untuk mengendalikan 2 motor DC sebagai *axis* dan 1 tombol untuk penggerak pelatuk tembak. Pada Android juga menampilkan hasil gambar yang ditangkap oleh kamera. *Micro controller* digunakan sebagai pusat pengolahan data.

### **1.2 Rumusan Masalah**

Sesuai dengan latar belakang yang telah diuraikan diatas, maka perumusan masalah dalam penelitian ini adalah :

- a. Bagaimana merancang sistem android sebagai remote kendali tembak?
- b. Bagaimana cara pembuatan dan perancangan program penerima sinyal dari android untuk menggerakkan motor?

### **1.3 Batasan Masalah**

Untuk menghindari pembahasan yang meluas maka ditentukan batasan masalah sebagai berikut :

- a. Orientasi senjata memiliki 2 derajat kebebasan yaitu arah Horizontal  $360^\circ$  dan Vertical  $60^\circ$ .
- b. Penghubung alat dan kontrol dengan wifi.

### **1.4 Tujuan**

Tujuan utama dari tugas akhir ini adalah untuk merealisasikan suatu alat tembak yang dikendalikan jarak jauh menggunakan Android. Dari uraian tersebut, maka dapat dibagi menjadi beberapa tujuan dalam proyek akhir ini, yaitu :

- Pengendali motor DC dengan android sebagai remote.
- Sebagai alat dalam pertahanan negara.
- Mengurangi korban jiwa dalam peperangan.

### **1.5 Metodologi Penelitian**

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu : tahap studi pustaka dan survey awal, tahap perencanaan dan pembuatan alat, tahap Perencanaan dan pembuatan software, tahap ujia coba dan analisis data, dan yang terakhir adalah penyusunan laporan berupa buku Tugas Akhir.

### **1.6 Sistematika Laporan**

Pembahasan Tugas Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

#### **Bab I           Pendahuluan**

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

#### **Bab II          Teori Dasar**

Bab ini menjelaskan dasar teori yang berisi tentang konsep yang dijadikan landasan dan mendukung dalam perencanaan serta pembuatan alat yang dibuat.

#### **Bab III       Perancangan dan Pembuatan Alat**

Bab ini membahas tentang perencanaan dan pembuatan perangkat keras (*hardware*) yang terdiri atas rangkaian elektronika, desain mekanik serta perangkat lunak (*software*) yang terdiri atas program yang akan digunakan untuk menjalankan alat tersebut.

#### **Bab IV      Pengukuran dan Pengujian**

Bab ini membahas tentang pengujian alat dan analisa data yang didapat dalam pengujian alat.

#### **Bab V      Penutup**

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

### **1.7 Relevansi**

Dengan adanya tugas akhir ini diharapkan dengan pengembangan lebih lanjut dapat dijadikan sebagai referensi atau ide dalam kemajuan pertahanan, khususnya sistem pertahanan batasan negara.

-----Halaman ini sengaja di kosongkan-----



## BAB II

### TEORI DASAR

Awal mula tembak atau senjata api pertama kali dibuat tahun 1800-an yang digunakan dalam perang dunia dan masih manual. Dari perkembangannya tembak menjadi senjata paling mematikan. Tahun 1860-an teknologi tembak lebih berkembang yang sebelumnya manual menjadi semi *automatic* atau *assault rifle*. Di era tahun 90-an senjata lebih pberkembang lagi dengan adanya senjata machine gun dan guttling gun. Namun senjata tersebut untuk peperangan yang jaraknya tidak terlalu jauh maka dalam perang akan banyak menimbulkan korban jiwa yang tidak sedikit. Terdapat juga senjata yang memiliki jara yang jauh yaitu sniper tetapi sniper sulit digunakan untuk peperangan dalam jumlah pasukan yang banyak.

Maka dengan adanya perkembangan senjata seperti itu perlulah teknologi senjata yang lebih maju dan dapat mengurangi timbulnya korban jiwa. Tembak kendali jarak jauh yaitu senjata yang dapat di operasikan dari jarak jauh dengan meggunakan remote kontrol sebagai kontrol untuk menggerakkan senjata. Dengan adanya alat tembak yang seperti ini dapat memberikan tingkat berkurangnya jatuh korban dalam peperangan, dalam hal ini kendali senjata dapat di remote dengan android yang dokoneksikan dengan *micro controller* menggunakan wifi. Sistem pada penembak sendiri terdapat 2 derajat kebebasan sebagai arah kendali tembak dan terdapat kamera sebagai pengganti indra mata dan arah tembak dapat di posisikan pada sasaran dengan dikendalikan android sebagai *remote control*. Dalam bab ini akan dibahas mengenai teori – teori yang berkaitan dengan peralatan yang akan dirancang. Teori yang mendukung penyelesaian tugas akhir ini diantaranya adalah mengenai; Motor DC, Driver Motor L298, dan *micro controller*.

#### 2.1 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 pin *input* dari *digital output* dimana 6 pin *input* tersebut dapat digunakan sebagai *output PWM* dan 6 pin *analog input*, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP *header*, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board Arduino Uno* ke komputer dengan menggunakan

kabel USB atau listrik dengan AC yang ke adaptor DC atau baterai untuk menjalankannya.

Uno berbeda dengan semua *board* sebelumnya dalam hal koneksi USB-*to*-serial yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter USB-*to*-serial berbeda dengan *board* sebelumnya yang menggunakan *chip* FTDI driver USB-*to*-serial.

Nama “Uno” berarti satu dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian *board* USB Arduino, dan sebagai model referensi untuk *platform* Arduino, untuk perbandingan dengan versi sebelumnya, lihat indeks *board* Arduino pada poin 2.5.1. untuk *board* arduino lihat pada Gambar 2.10



**Gambar 2.1** Arduino UNO

### 2.1.1 Ringkasan

1. *Microcontroller* ATmega328
2. Operasi dengan daya 5Volt
3. *Input* Tegangan (disarankan) 7-12V
4. *Input* Tegangan (batas) 6-20V
5. *Digital* I / O Pin 14 (dimana 6 memberikan *output* PWM)
6. *Analog Input* Pin 6
7. DC Lancar per I / O Pin 40 mA
8. Saat 3,3V Pin 50 mA DC
9. *Flash Memory* 32 KB (ATmega328) yang 0,5 KB digunakan oleh *bootloader*
10. SRAM 2 KB (ATmega328)
11. EEPROM 1 KB (ATmega328)
12. *Clock Speed* 16 MHz

### 2.1.2 Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya *eksternal* (otomatis).

*Eksternal* (non-USB) daya dapat berasal baik dari AC-ke adaptor-DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan *plug jack* pusat-positif ukuran 2.1 mm konektor *POWER*. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin pin *header* dari konektor *POWER*.

Kisaran kebutuhan daya yang disarankan untuk *board* Uno adalah 7 sampai dengan 12 volt, jika diberi daya kurang dari 7 volt kemungkinan pin 5v Uno dapat beroperasi tetapi tidak stabil kemudian jika diberi daya lebih dari 12V, regulator tegangan bisa panas dan dapat merusak *board* Uno.

Pin listrik adalah sebagai berikut:

1. VIN. Tegangan masukan kepada *board* Arduino ketika itu menggunakan sumber daya eksternal (sebagai pengganti dari 5 volt koneksi USB atau sumber daya lainnya).
2. 5V. Catu daya digunakan untuk daya mikrokontroler dan komponen lainnya.
3. 3v3. Sebuah pasokan 3,3 volt dihasilkan oleh *regulator on-board*.
4. GND. *Ground* pin.

### 2.1.3 Memori

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk *bootloader*), 2 KB dari SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan EEPROM *library*).

### 2.1.4 Input dan Output

Masing-masing dari 14 pin digital di Uno dapat digunakan sebagai *input* atau *output*, dengan menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*, beroperasi dengan daya 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull-up* resistor (secara default terputus) dari 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus:

1. Serial: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin ini dihubungkan ke pin yang berkaitan dengan *chip* Serial ATmega8U2 USB-to-TTL.
2. Eksternal menyela: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik

atau turun, atau perubahan nilai. Lihat (*attachInterrupt*) fungsi untuk rincian lebih lanjut.

3. PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite()*.
4. SPI: 10 (SS), 11 (Mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
5. LED: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai *HIGH*, LED on, ketika pin bernilai *LOW*, LED off.

Uno memiliki 6 masukan analog, berlabel A0 sampai dengan A5, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda). Selain itu, beberapa pin memiliki fungsi khusus:

1. I2C: A4 (SDA) dan A5 (SCL). Dukungan I2C (TWI) komunikasi menggunakan perpustakaan *Wire*.
2. Aref. Tegangan referensi (0 sampai 5V saja) untuk *input* analog. Digunakan dengan fungsi *analogReference()*.
3. *Reset*. Bawa baris ini *LOW* untuk me-*reset* mikrokontroler.

### 2.1.5 Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi serial melalui USB dan sebagai *port virtual com* untuk perangkat lunak pada komputer. *Firmware* '8 U2 menggunakan *driver* USB standar COM, dan tidak ada *driver* eksternal yang diperlukan. Namun, pada Windows diperlukan, sebuah *file* inf. Perangkat lunak Arduino terdapat monitor serial yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari *board* Arduino. LED RX dan TX di papan tulis akan berkedip ketika data sedang dikirim melalui *chip* USB-to-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1).

Sebuah *Software Serial library* memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada *board* Uno. ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino memiliki perpustakaan yang dapat untuk menyederhanakan penggunaan bus I2C, lihat dokumentasi untuk rincian. Untuk komunikasi SPI, menggunakan perpustakaan SPI.

## 2.2 Modul Wifi ESP8266

ESP8266 adalah sebuah modul WiFi yang akhir-akhir ini semakin digemari para hardware developer. Selain karena harganya yang sangat terjangkau, modul WiFi serbaguna ini sudah bersifat SoC (System on Chip), sehingga kita bisa melakukan programming langsung ke ESP8266 tanpa memerlukan mikrokontroler tambahan. Kelebihan lainnya, ESP8266 ini dapat menjalankan peran sebagai adhoc akses poin maupun klien sekaligus. ESP8266 dikembangkan oleh pengembang asal negeri tiongkok yang bernama “Espressif”. Produk seri ESP8266 memiliki banyak sekali varian. Salah satu varian yang paling sering kita jumpai adalah ESP8266 seri ESP-01.

Spesifikasi Umum ESP8266:

- 802.11 b/g/n
- Integrated low power 32-bit MCU
- Integrated 10-bit ADC
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLL, regulators, and power management units
- Supports antenna diversity
- WiFi 2.4 GHz, support WPA/WPA2
- Support STA/AP/STA+AP operation modes
- Support Smart Link Function for both Android and iOS devices
- SDIO 2.0, (H) SPI, UART, I2C, I2S, IR Remote Control, PWM, GPIO
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4s guard interval
- Deep sleep power <10uA, Power down leakage current < 5uA
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)
- +20 dBm output power in 802.11b mode
- Operating temperature range -40C ~ 125C
- FCC, CE, TELEC, WiFi Alliance, and SRRC certified

Pada umumnya, ESP8266 dapat diprogram dengan:

melalui AT command via serial komunikasi UART pemrograman ke mikrokontroler yang ada di ESP8266 menggunakan Arduino IDE dengan Core yang sudah terinstall ESP8266.

Kelebihan lain ESP8266 adalah memiliki deep sleep mode, sehingga penggunaan daya akan relatif jauh lebih efisien dibandingkan dengan modul WiFi.

### 2.3 Motor DC

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak langsung/direct-unidirectional. Motor DC memiliki 3 bagian atau komponen utama untuk dapat berputar sebagai berikut:

1. Kutub medan

Kutub medan. Motor DC sederhana memiliki dua kutub medan: kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi ruang terbuka diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet.

2. *Current elektromagnet*

*Current Elektromagnet* atau Dinamo. Dinamo yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, dinamo berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi.

3. *commutator*

*Commutator*. Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk transmisi arus antara dinamo dan sumber daya.

Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur:

Tegangan dinamo : meningkatkan tegangan dinamo akan meningkatkan kecepatan

Arus medan : menurunkan arus medan akan meningkatkan kecepatan.

Hubungan antara kecepatan, flux medan dan tegangan dinamo ditunjukkan dalam persamaan berikut:

Gaya Elektromagnetik (E)

$$E = K\Phi N$$

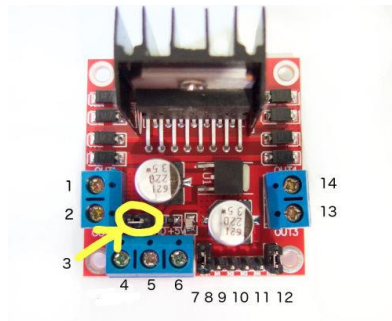
$$T = K\Phi I_a$$

Dimana:

- $E$  = gaya elektromagnetik yang dikembangkan pada terminal dinamo (volt)
- $\Phi$  = flux medan yang berbanding lurus dengan arus medan
- $N$  = kecepatan dalam RPM (putaran per menit)
- $T$  = torque elektromagnetik
- $I_a$  = arus dinamo
- $K$  = konstanta persamaan

## 2.4 Modul L298N

Modul driver L298N, menggunakan chip ST L298N yang dapat secara langsung mengontrol dua motor DC 3-30V, dan menyediakan keluaran 5 V, modul dapat mengontrol dengan mudah kecepatan dan Arah pergerakan Motor DC.



**Gambar 2.2** Modul L298N

Keterangan :

1. DC motor 1 “+” atau stepper motor A+
2. DC motor 1 “-” atau stepper motor A-
3. 12V jumper – lepaskan jumper ini jika menggunakan sumber lebih dari 12 V DC. Ini memungkinkan sumber dari regulator pada Arduino 5V.
4. Hubungkan sumber tegangan motor disini, maksimum 35 V DC. Lepaskan 12V jumper jika  $>12V$ .
5. GND
6. 5V output jika 12V jumper digunakan, ideal untuk mensuplai Arduino Anda(etc).

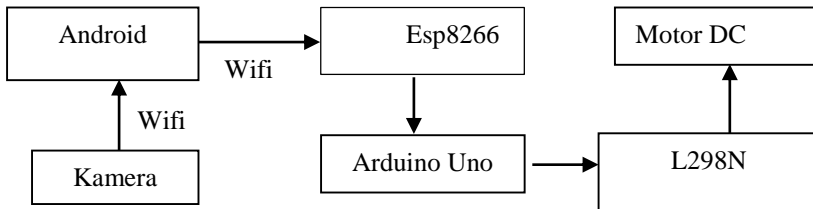
7. DC motor 1 enable jumper. Lepaskan ini ketika menggunakan stepper motor. Hubungkan ke keluaran PWM untuk menatur kecepatan motor.
8. IN1
9. IN2
10. IN3
11. IN4
12. DC motor 2 enable jumper. Lepaskan ini ketika menggunakan stepper motor. Hubungkan ke keluaran PWM untuk menatur kecepatan motor.
13. DC motor 2 “+” or stepper motor B+
14. DC motor 2 “-” or stepper motor B-



### BAB III

## PERANCANGAN DAN PEMBUATAN ALAT

Bab ini berisi tahapan yang dilakukan dalam perencanaan dan pembuatan tugas akhir. Penjelasan diawali dengan penjelasan blok fungsional sistem secara keseluruhan, kemudian perancangan perangkat keras dan diakhiri dengan perangkat lunak.

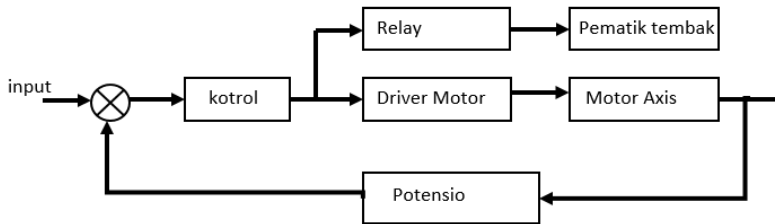


**Gambar 3.1** Diagram Blok Fungsi Cara Kerja Alat

Untuk mendapatkan penembak portabel yang berjalan sesuai harapan diperlukan bagian pokok, yaitu:

1. Motor DC, berfungsi sebagai aktuator penggerak axis dengan mendapat sinyal PWM dari pengendali.
2. Minimum sistem arduino uno, mikrokontroler yang berfungsi sebagai pengendali, pengolah sinyal masuk dan keluar. Pin yang digunakan adalah pin *digital input*, PWM, pin *analog*.
3. Sensor sudut, sensor ini terdiri dari potensio atau sensor resistif dengan pengaturan pembagian tegangan.

Adapun penjelasan dari diagram blok pada Gambar 3.1 dan juga tahapan untuk menggerakkan penembak portabel oleh motor DC sebagai penggerak axis vertikal dan horizontal derajat tertentu. Penjelasan dari masing-masing bagian dijelaskan pada sub bab pembuatan perangkat elektronik, pembuatan perangkat mekanik, dan pembuatan perangkat lunak.



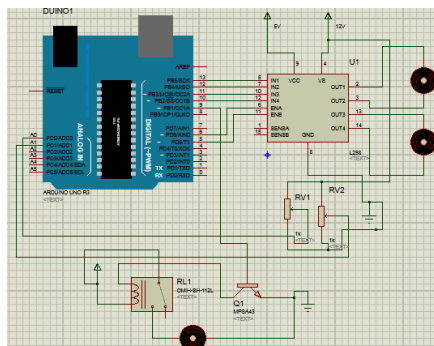
**Gambar 3.2** Blok Diagram Kontrol

Pada blok diagram kontrol input didapat dari android yang mengim data dan diterima pada esp8266 dan data dibaca dan di olah pada arduino. Bila kontrol mendapat input untuk menggerakkan motor maka kontrol akan memberi sinyal pada driver motor untuk menggerakkan motor sesuai dengan perintah axis yang diberikan bisa kanan, kiri, atas, atau bawah. Bila data input yang diterima dari android untuk menggerakkan pematik maka kontrol akan megaktifkan *relay* agar pematik tembak aktif. Untuk sensor digunakan ketika motor berputar mencapai batas sudut yang diseting maka motor akan berhenti dan motor tidak bisa bergerak kecuali bergerak ke arah berlawanan arah dari gerakan awal.

### 3.1 Pembuatan Perangkat Elektronik

Dalam perangkat elektronik, terdapat beberapa elemen yang harus disusun untuk dapat menggerakkan motor DC dengan baik. Elemen-elemen tersebut adalah sebagai berikut:

#### 3.1.1 Skematik *Hardware* Kontrol



**Gambar 3.3** Skematik *Hardware* Kontrol

Pada skematik *hardware* kontrol di atas yaitu penggabungan semua komponen penyusun sistem yang akan di kendalikan oleh arduino sebagai *micro controller*.

### 3.1.2 Pemilihan Jenis Motor DC

Motor DC berfungsi sebagai aktuator untuk menggerakkan axis vertikan dan horizontal. Untuk penggerakan vertikal menggunakan motor DC yang menggunakan *gearbox* menyilang.



**Gambar 3.4** *Gearbox* Menyilang

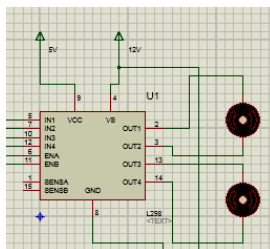
Pada gearbox menyilang memiliki sifat yang mengunci bila motor berhenti bergerak. Maka dengan menggunakan motor dengan gearbox menyilang dapat membuat penguncian pergerakan secara mekanik bila motor DC telah mati atau sudah tidak mendapat sumber tegangan. Untuk motor DC sebagai penggerak yang horizontal menggunakan motor gearbox sejajar karena pada penggerak horizontal motor hanya berbeban pada penggerak kanan dan kiri yang tanpa harus menahan beban ketika berhenti seperti penggerak vertikal. Pada pemilihan torsi motor DC untuk penggerak vertikal harus memiliki torsi yang besar karena menggerakkan tembak keatas membutuhkan tenaga yang lebih besar. Bila torsi pada motor vertikal tidak memiliki torsi yang besar maka motor DC hanya akan bergetar dengan kondisi diam yang mengakibatkan terjadinya kebakaran pada motor DC, karena pada penggerak vertikal memiliki menahan beban yang berat.



**Gambar 3.5** Gearbox sejajar

### 3.1.3 Pengkabelan Axis Penggerak Motor DC dan L298N

Motor DC berfungsi sebagai aktuator untuk menggerakkan *axis* vertikal dan horizontal. Pin PWM dihubungkan dengan pin PWM pada arduino. Pin PWM dari pengendali ke driver motor berfungsi untuk mengirimkan sinyal PWM ke motor DC dapat bergerak sesuai dengan kecepatan yang di inginkan.

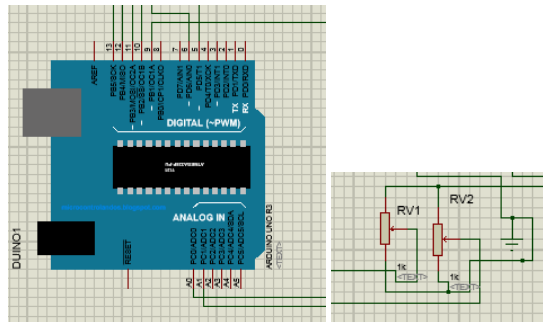


**Gambar 3.6** Pengkabelan Axis Penggerak Motor DC dan L298N

### 3.1.4 Pengkabelan Minimum Sistem dengan Sensor Resistif Sebagai Pengukur Sudut.

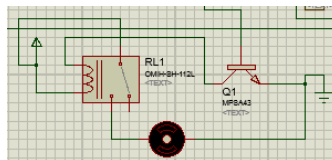
Rangkaian sensor pengatur sudut terdiri dari resistor variabel (potensiometer) 50k $\Omega$ . Potensiometer dipasang pada gigi gear yang terpasang pada penggerak motor dengan pengambilan gradien pada sudut minimal ke maksimal. Nilai gradien pada sudut tersebut digunakan sebagai parameter nilai ADC pada program minimum sistem. Pengaturan sudut berfungsi sebagai pengamanan pada putaran axis maka ketika data pada ADC mencapai nilai minimal atau maksimal maka axis akan berhenti, untuk menggerakkan axis kembali dengan cara membalik arah putar pada axis. Pengkabelan pada sensor dimasukkan pada pin Analog(A0, A1). Sensor bawah atau gerak horizontal dimasukkan pada pin A0 diberi batasan derajat 0° sampai 360° dan untuk

sensor atas atuk gerak vertikal dimasukkan pada pin A1 dan diberi batasan derajat  $0^{\circ}$  sampai  $60^{\circ}$ .



**Gambar 3.7** Pengkabelan Sistem Arduino Uno dengan Sensor Resistif

### 3.1.5 Pengkabelan pada Pematik Tembak



**Gambar 3.8** Pengkabelan pada Pematik Tembak

Pada pematik tembak menggunakan motor DC sebagai pematik tembak dan yang di rangkai dengan relay dan transistor npn karena sumber untuk tenatik tembak secara terpisah dan tidak berbalikan arah seperti pengatur axis. Transistor basis masuk pada pin 9 arduino, koil relay mengambil sumber tegangan dari batrai keluaran relay masuk ke kolektor transistor dan emitor transistor masuk ke *ground* seperti pada gambar 3.7.

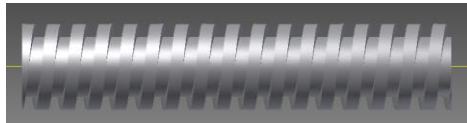
## 3.2 Pembuatan Perangkat Mekanik

Dalam perangkat mekanik, terdapat beberapa bagian yang harus dibuat untuk dapat bekerja dengan baik. Terdapat 2 bagian besar dalam perangkat mekaniknya yang terdiri dari 1 buah dudukan tembak atas untuk axis vertikal dan 1 buah meja penyangga dudukan tembak. Berikut tahapan-tahapan pembuatan perangkat mekaniknya:

### 3.2.1 Pembuatan Dudukan Atas

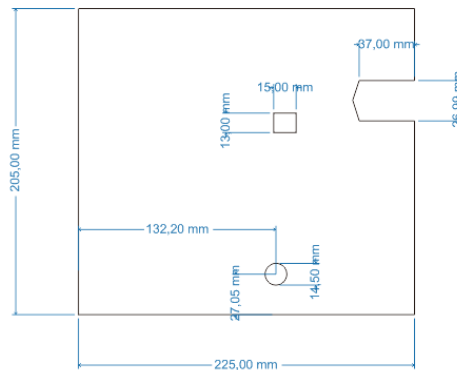
Bagian dasar dari tembak yaitu: dudukan tembak bagian atas , penjepit tembak, poros inti putar, dudukan tembak bagian bawah. Bahan yang

digunakan untuk membuat dudukan atas ini terbuat dari bahan besi dan untuk penjepit tembak dari bahan akrilik, dalam hal ini pemilihan bahan dengan menggunakan besi karena besi memiliki masa yang brat dan kuat. Maka dengan menggunakan besi alat akan tidak mudah patah dan bergeser ketika penatik tembak aktif, dan untuk penjepit tembak menggunakan akrilik agar beban pada penggerak vertikal sedikit ringan dan bahan dari akrilik memiliki ketahanan yang cukup kuat. Untuk penghubung penjepit tembak dan dudukan atas menggunakan as ulir dan baut berdiameter 14 mm dengan panjang 22 cm dan untuk menghubungkan penyangga bawah dan penyangga atas menggunakan as ulir berdiameter 16 mm dan baut sebagai pemberi ketahanan ketika di gabung agar tidak mudah lepas.



**Gambar 3.9** Poros Inti Putar

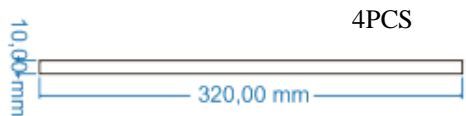
Gambar 3.4 adalah poros inti untuk memudahkan penggerakan tembak secara vertikal yang berfungsi sebagai penghubung antara dudukan senjata yang di hubungkan dengan dudukan atas senjata. Selain itu, pada bagian dasar terdapat dudukan untuk tembak terdapat beberapabagian :



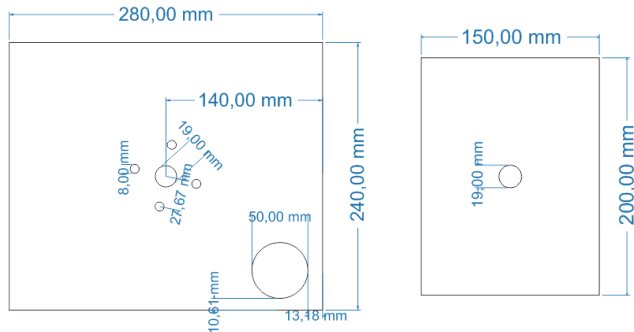


**3.2.2 Dudukan bagian bawah**

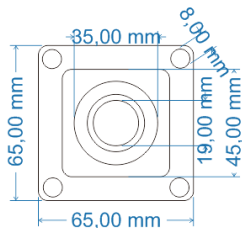
Bagian selanjutnya adalah bagian dudukan bawah atau kaki penyangga yang secara tidak langsung memiliki 2 buah lubang atas dan bawah yang terdapat bearing yang difungsikan untuk mempermudah pergerakan tembak secara horizontal.



**Gambar 3.12** kaki penyangga

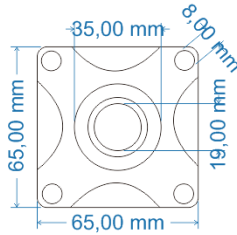


**Gambar 3.13** Bagian Penyangga Atas dan Bawah

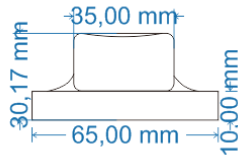


**Gambar 3.14** Bearing Tampak Bawah

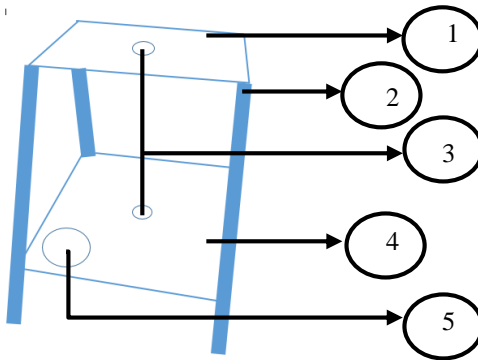




**Gambar 3.15** Bearing tampak Atas



**Gambar 3.16** Bearing Tampak Samping



**Gambar 3.17** Penggabungan Penyangga Bawah

Keterangan :

1. Plat bagian atas seperti pada gambar 3.13.
2. Kaki penyangga seperti pada gambar 3.12.
3. Lubang menancapkan as ulir dan di beri bearing seperti pada gambar 3.14, 3.15, 3.16.
4. Plat bagian bawah seperti pada gambar 3.13.
5. Lubang untuk menaruh motor penggerak horizontal.

Dari gambar 3.17 memiliki lubang di tengahnya yang digunakan untuk masukkan as ulir dari penyangga atas dan disetiap lubang yang ditengah di pasang bearing seperti pada gambar 3.14, 3.15, 3.16 agar penggerakan axis horizontal lebih mudah dan ringan karena bearing memiliki fungsi untuk mengurangi gaya gesek pada medan berputar dengan poros.

### **3.3 Pembuatan Perangkat Lunak**

Dalam perangkat lunak, terdapat beberapa program yang harus dibuat untuk dapat menggerakkan motor DC dengan baik. Tahapan pembuatan tersebut adalah sebagai berikut:

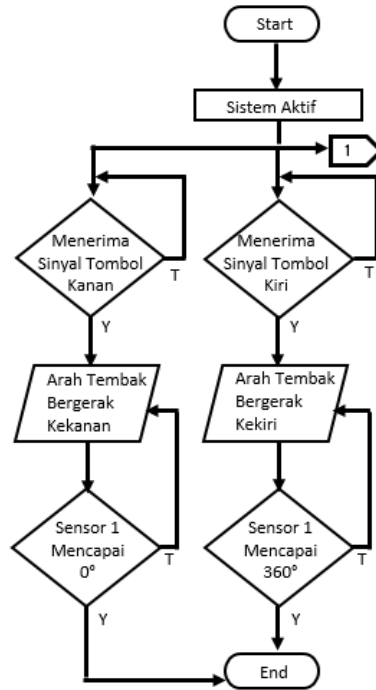
#### **3.3.1 Pembuatan *Flowchart* Program**

*Flowchart* merupakan bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Bagan ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung. *Flowchart* ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan poses kegiatan menjadi lebih jelas.

Untuk pengolahan data dengan komputer, dapat dirangkum urutan dasar untuk pemecahan suatu masalah, yaitu;

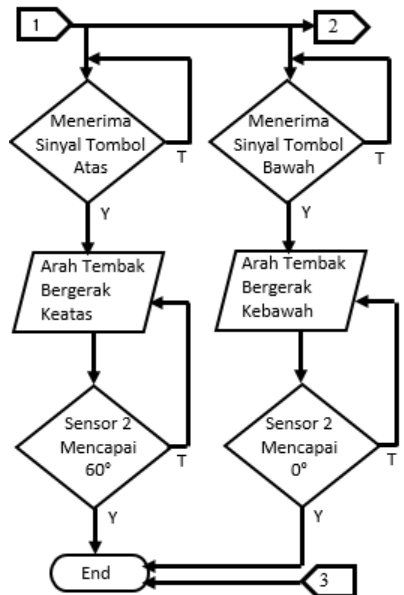
- *START*: berisi instruksi untuk persiapan peralatan yang diperlukan sebelum menangani pemecahan masalah.
- *READ*: berisi instruksi untuk membaca data dari suatu peralatan.
- *PROCESS*: berisi kegiatan yang berkaitan dengan pemecahan persoalan sesuai dengan data yang dibaca.
- *WRITE*: berisi instruksi untuk merekam hasil kegiatan ke peralatan *output*.
- *END*: mengakhiri kegiatan pengolahan

*Flowchart* program dari Tugas Akhir ini meliputi seluruh sistem jalannya alat ini. Sistem yang dimaksud adalah sistem umum secara keseluruhan.



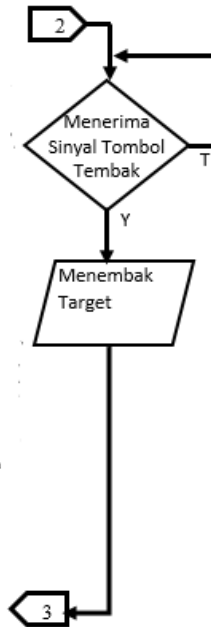
**Gambar 3.18** *Flowchart* Gerak Motor 1

*Flowchart* ini menjelaskan tentang penerimaan data untuk menggerakkan motor DC bagian bawah atau untuk *axis* horizontal dengan cara ketika menerima data maka motor bergerak dan bila mencapai batas sensor yang ditentukan bisa  $0^{\circ}$  atau  $360^{\circ}$  maka motor akan berhenti dan hanya bisa di jalankan berbalik arah dari arah sebelumnya.



**Gambar 3.19** *Flowchart* Gerak Motor 2

*Flowchart* ini menjelaskan tentang penerimaan data untuk menggerakkan motor DC bagian bawah atau untuk *axis* vertikal dengan cara ketika menerima data maka motor bergerak dan bila mencapai batas sensor yang ditentukan bisa  $0^{\circ}$  atau  $60^{\circ}$  maka motor akan berhenti dan hanya bisa di jalankan berbalik arah dari arah sebelumnya.



**Gambar 3.20** *Flowchart* Pelatuk Tembak

Pada *flowchart* ini ketika arduino menerima sinyal perintah dari android untuk menembak, maka arduino akan mengaktifkan pelatuk tembak dan akan menebbak sasaran yang telah diarahkan sebelumnya. Untuk melihat *flowchart* gabungan dapat dilihat di lampiran B.*flowchart*.

### 3.3.2 Pendefinisian Pin dan Pustaka

```

const int enb = 6;
const int ena = 5;
  
```

**Gambar 3.21** Program Definisi Pustaka dan Tipe Data

Dalam rancangan program yang dibuat, dimasukkan pustaka untuk kendali motor DC yang telah terhubung dengan driver motor dan disediakan oleh Arduino IDE. Setelah memasukan pustaka driver motor, diberi nama setiap motor DC yang akan digunakan dalam pemrograman, dalam hal ini

untuk motor 1 diberi nama ena(PWM), in1, in2 dan motor 2 diberi nama enb(PWM), in3, in4.

### 3.3.3 Segmen Program untuk Pengaturan Awal

```
void setup() {  
    Serial.begin(9600);  
    wifi.begin(9600);  
  
    pinMode(enb, OUTPUT);  
    analogWrite(6, 220);  
    pinMode(ena, OUTPUT);  
    analogWrite(5, 150);  
    pinMode(10, OUTPUT);  
    digitalWrite(10, HIGH);  
    pinMode(11, OUTPUT);  
    digitalWrite(11, HIGH);  
    pinMode(12, OUTPUT);  
    digitalWrite(12, HIGH);  
    pinMode(13, OUTPUT);  
    digitalWrite(13, HIGH);  
    pinMode(9, OUTPUT);  
    digitalWrite(9, LOW);  
}
```

**Gambar 3.22** Segmen Program Untuk Pengaturan Awal

*Void* pada Arduino berfungsi sebagai kata kunci untuk membuat subprogram dari program utama. Jika fungsi *void* digunakan sebagai *void setup*, maka subprogram tersebut berfungsi sebagai program awal dari Arduino dan tidak diulang saat program dieksekusi. Dari masing-masing subprogram terdapat fungsi *digital write* dan *seral print* yang hanya dapat dikendalikan apabila mendapat masukan melalui serial dalam program.

Pada gambar 3.22, terdapat variabel *a* yang dinyatakan sebagai *ena* dan *enb* untuk PWM, *in1* sampai *in4*. Hal ini berarti bahwa pada awal program, *a* didefinisikan sebagai pengatur pin ke driver motor. Kemudian fungsi *pinMode* berfungsi untuk merepresentasikan fungsi pin yang akan dipanggil. Jika pada potongan program gambar 3.22 tertulis *pinMode(pinOutput, OUTPUT)*, maka fungsi tersebut menyatakan bahwa pin yang dinamakan sebagai *pinOutput* berfungsi sebagai *OUTPUT*. Sebagaimana sebelumnya *pinOutput* adalah nama untuk pin yang digunakan dalam potongan program ini.

### 3.3.4 Segmen Program untuk Menjalankan motor DC

```
void loop() {  
  // Cek jika ESP mengirimkan pesan  
  if(wifi.available()) {  
  
    if(wifi.find("+IPD,")) {  
      delay(100);  
      connectionId = wifi.read()-48;  
      wifi.find("pin=");  
      pinNumber = (wifi.read()-48)*9;  
      pinNumber += (wifi.read()-48);  
  
      secondNumber = (wifi.read()-48);  
      if(secondNumber>=0 && secondNumber<=9)  
      {  
        pinNumber*=9;  
        pinNumber +=secondNumber;  
      }  
      pinMode(pinNumber,OUTPUT);  
      digitalWrite(pinNumber, !digitalRead(pinNumber));  
    }  
  }  
}
```

**Gambar 3.23** Segmen Program Untuk Menjalankan Motor DC

Pada gambar 3.23, terdapat fungsi *void loop* yang diikuti oleh nama subprogram yaitu menjalankan motor1 ,menjalankan motor 2, dan menjalankan pematik tembak. Pada void loop ini difungsikan untuk program di lakukan panggilan secara berulang ulang.

-----Halaman ini sengaja di kosongkan-----



## **BAB IV**

### **PENGUKURAN DAN PENGUJIAN**

Untuk mengetahui bahwa alat telah bekerja dengan benar maka perlu dilakukan pengujian alat yang meliputi pengujian perangkat keras dan pengujian perangkat lunak. Pengujian yang dilakukan pada peralatan untuk mengetahui kesesuaian antara teori dengan hasil perancangan, yaitu dengan mengetahui hasil pengukuran pada setiap perangkat yang telah dibuat. Alat yang telah selesai dapat dilihat pada Gambar 4.1



**Gambar 4.1** Gambar Keseluruhan Alat

#### **4.1 Pengukuran dan Pengujian Arduino dengan L298N dan Motor DC**

Tujuan dari pengambilan data dari kecepatan gerak motor DC adalah untuk mengetahui tingkat akurasi tembakan agar lebih presisi dalam memosisikan sasaran pada target.

Untuk melakukan pengukuran gerakan motor DC, maka dilakukan pemrograman melalui Arduino dengan *interval* PWM dengan bervariasi di setiap tegangan yang berbeda hingga mendapat hasil yang lebih sesuai atau yang diinginkan.

**Tabel 4.1** Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 12 Volt

No.	Tegangan	PWM	Gerakan	
			Vertikal	Horizontal
1.	12	255	2	3
2.	12	225	1	3
3.	12	200	0	2
4.	12	180	0	2
5.	12	160	0	2
6.	12	140	0	1
7.	12	120	0	0
8.	12	100	0	0
9.	12	80	0	0

**Tabel 4.2** Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 15 Volt

No.	Tegangan	PWM	Gerakan	
			Vertikal	Horizontal
1.	15	255	2	3
2.	15	225	2	3
3.	15	200	2	3
4.	15	180	2	3
5.	15	160	0	2
6.	15	140	0	1
7.	15	120	0	1
8.	15	100	0	0
9.	15	80	0	0

**Tabel 4.3** Data Pengukuran Gerak Vertikal dan Horizontal Motor DC pada Tegangan 17 Volt

No.	Tegangan	PWM	Gerakan	
			Vertikal	Horizontal
1.	17	255	3	3
2.	17	225	3	3
3.	17	200	3	3
4.	17	180	3	3
5.	17	160	2	3
6.	17	140	2	2
7.	17	120	0	2
8.	17	100	0	1
9.	17	80	0	1

Keterangan kode nilai vertikan dan horizontal:

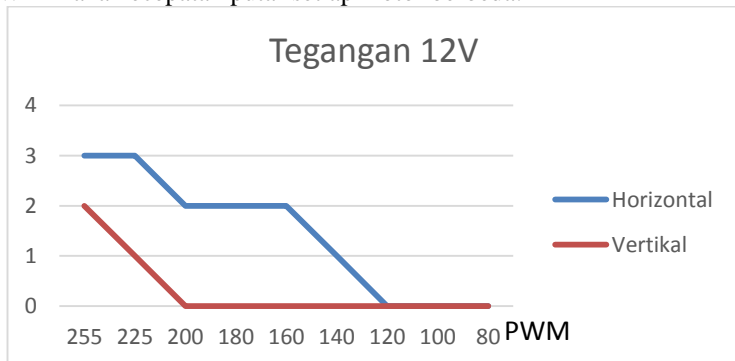
0 : tidak bergerak

1 : bergerak lambat

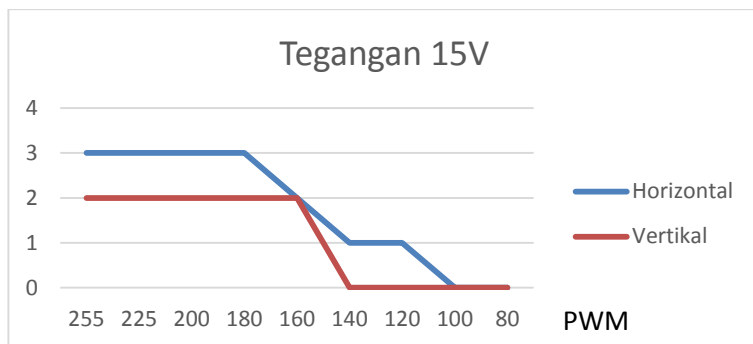
2 : bergerak sedang

3: bergerak cepat

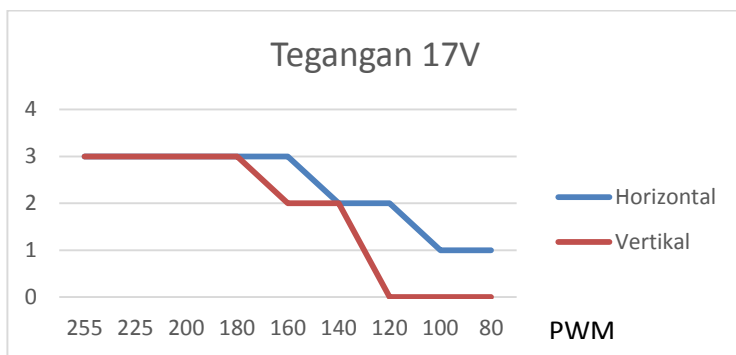
Pada percobaan yang dilakukan, didapat bahwa masing-masing motor DC memiliki tingkat respon kecepatan yang berbeda semakin tinggi tegangan dan PWM maka kecepatan putar setiap motor berbeda.



**Gambar 4.2** Gambar Grafik Gerakan Vertikal dan Horizontal Pada Tegangan 12 Volt



**Gambar 4.3** Gambar Grafik Gerakan Vertikal dan Horizontal pada Tegangan 15 Volt



**Gambar 4.4** Gambar Grafik Gerakan Vertikal dan Horizontal pada Tegangan 17 Volt

#### 4.2 Pengujian Respon Kontrol Motor DC

Pada pengujian ini akan dilihat respon dari pergerakan motor DC berdasarkan masukkan dari Aplikasi kontrol pada android.

**Tabel 4.4** Respon Kontrol Motor DC

Jarak	Respon Motor DC
1 Meter	Delay 0,5 Detik
3 Meter	Delay 0,5 Detik
6 Meter	Delay 0,5 Detik
8 Meter	Delay 1 Detik

**Tabel 4.4** Respon Kontrol Motor DC

Jarak	Respon Motor DC
10 Meter	Delay 1,5 Detik
15 Meter	Delay 1,5 Detik
25 Meter	Disconnect

#### 4.3 Pengujian Respon Penerimaan Data Ke Motor DC secara Berurutan

Pengujian ini bertujuan untuk melihat seberapa besar delay yang dibutuhkan Aplikasi saat mengirim data serta seberapa besar respon pada motor DC dapat bekerja secara normal.

**Tabel 4.5** Respon Motor DC Diberi Input Secara Berurutan dalam jarak 6 meter

Delay		Arah 1	Arah 2	Arah 3	Arah 4	Arah 5
0,2 detik	Input	kanan	Kiri	Atas	Kanan	Bawah
	Respon	Mati	Mati	Mati	Mati	Mati
0,5 detik	Input	Kiri	Kanan	Bawah	Kiri	Atas
	Respon	Aktif	Mati	Mati	Mati	Aktif
0,7 detik	Input	Kiri	Kanan	Bawah	Atas	Kanan
	Respon	Aktif	Mati	Mati	Aktif	Mati
1 detik	Input	Bawah	Kiri	Kanan	Atas	Kiri
	Respon	Aktif	Mati	Aktif	Mati	Aktif
1,3 detik	Input	Atas	Kanan	Bawah	Kiri	Kanan
	Respon	Aktif	Mati	Aktif	Mati	Aktif
1,5 detik	Input	Kiri	Atas	Kanan	Bawah	Kiri
	Respon	Aktif	Mati	Aktif	Mati	Aktif
2 detik	Input	Atas	Bawah	Kiri	Atas	Kanan
	Respon	Aktif	Aktif	Aktif	Aktif	Aktif
2,5 detik	Input	Kiri	Kanan	Atas	Bawah	Kiri
	Respon	Aktif	Aktif	Aktif	Aktif	Aktif

Dari tabel 4.6 memiliki data tingkan respon dari pengirim dan yang diterima secara berurut dengan jeda waktu yang berbeda. Hasilnya semakin

kecil jeda waktu yang diterima dari android maka data tidak ada yang terbaca atau tingkat keberhasilan 0%. Jeda waktu dinaikkan menjadi 0,7 detik, data yang diterima dan terbaca ada yang aktif dan mati dan tingkat keberhasilannya 40%. Jeda waktu di naikkan 1,5 detik masih terdapat beberapa perintah tidak terbaca oleh arduino dan tingkat keberhasilan 60%. Pengujian dengan jeda waktu 2 detik memiliki tingkat keberhasilan 100%.

**Tabel 4.6** Respon Motor DC Diberi Input Secara Berurutan dalam jarak 15 meter

Delay		Arah 1	Arah 2	Arah 3	Arah 4	Arah 5
0,2 detik	Input	kanan	Kiri	Atas	Kanan	Bawah
	Respon	Mati	Mati	Mati	Mati	Mati
0,5 detik	Input	Kiri	Kanan	Bawah	Kiri	Atas
	Respon	Mati	Mati	Mati	Mati	Mati
0,7 detik	Input	Kiri	Kanan	Bawah	Atas	Kanan
	Respon	Mati	Mati	Mati	Mati	Mati
1 detik	Input	Bawah	Kiri	Kanan	Atas	Kiri
	Respon	Mati	Mati	Aktif	Mati	Mati
1,3 detik	Input	Atas	Kanan	Bawah	Kiri	Kanan
	Respon	Aktif	Mati	Aktif	Mati	Aktif
1,5 detik	Input	Kiri	Atas	Kanan	Bawah	Kiri
	Respon	Aktif	Mati	Mati	Aktif	Mati
2 detik	Input	Atas	Bawah	Kiri	Atas	Kanan
	Respon	Aktif	Mati	Aktif	Aktif	Mati
2,5 detik	Input	Kiri	Kanan	Atas	Bawah	Kiri
	Respon	Aktif	Aktif	Aktif	Aktif	Aktif
2,8 detik	Input	Atas	Kanan	Kiri	Bawah	Kanan
	Respon	Aktif	Aktif	Aktif	Aktif	Aktif

Dari hasil data pada tabel 4.7 semakin jauh jarak antara pengirim dan penerima maka *delay* yang dibutuhkan lebih besar.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari hasil pengujian dan pengukuran , dapat diambil kesimpulan bahwa:

1. Dari data pengukuran kecepatan gerak vertikal pada tegangan 12 volt dengan PWM 80 sampai 200 motor DC tidak dapat berputar atau bagerak, pada PWM 225 motor berputar lambat dan pada PWM 255 motor berputar sedang.
2. Dari data pengukuran kecepatan gerak horizontal pada tegangan 12 volt dengan PWM 80 sampai 120 motor DC tidak berputar, pada PWM 140 motor DC berputar lambat, pada PWM 160 sampai 200 motor berputar sedang dan pada PWM 225 sampai 255 motor berputar cepat.
3. Dari kesimpulan 1 dan 2 motor horizontal berputar lebih cepat dengan PWM yang sama dengan PWM motor vertikal karena motor penggerak vertikal lebih memiliki beban yang lebih berat.
4. Dalam jarak penerimaan data yang di kirim dari android menghasilkan data pengujian yang memiliki tingkat respon dari penerimaan data yang dibaca arduino, pada jarak 1 sampai 6 meter memiliki *delay* sebesar 0,5 detik, pada jarak 8 meter memiliki *delay* 1 detik, jarak 10 sampai 15 meter *delay* 1,5 detik dan jarak diatas 25 meter respon *disconnect*. Maka semakin jauh jarak antara pengirim dan penerima maka *delay* atau jeda waktu semakin besar.
5. Pada pengujian respon penerimaan data ke motor DC secara berurutan dengan jeda waktu yang berbeda maka semakin kecil jeda waktu yang diterima dari android maka data tidak ada yang terbaca atu tingkat keberhasilan 0%. Jeda waktu dinaikkan menjadi 0,7 detik, data yang diterima dan terbaca ada yang aktif dan mati dan tingkat keberhasilannya 40%. Jeda waktu di naikkan 1,5 detik masih terdapat beberapa perintah tidak terbaca oleh arduino dan tingkat keberhasilan 60%. Pengujian dengan jeda waktu 2 detik memiliki tingkat keberhasilan 100%.

#### **5.2 Saran**

Saran untuk penelitian selanjutnya adalah sebaiknya digunakan motor DC dengan torsi yang lebih besar dan tegangan yang lebih kecil karena pada penggerak vertikal memiliki beban yang sangat berat untuk menggerakkan

posisi tembak dan . Untuk penerimaan data disarankan untuk menggunakan modul wifi yang lebih bagus dari esp8266 karena pada esp8266 memiliki tingkat respon dan pengiriman data ke arduino memiliki *delay* yang sedikit besar maka hasil yang dicapai sedikit kurang memuaskan.



## DAFTAR PUSTAKA

- [1] Lin, Abney, and Bekey .2008.”Autonomous Military Robotics: Risk, Ethics, and Design”.
- [2] Andrew Caballero,Ian Stine,Steve Majo,Mike Bok.2008.”Auto-Targeting Sentry Gun with Friend/Foe Recognition”, Project Documentation University of Central Florida.
- [3] Shyam Sadasivan. 2006.”An Introduction to the ARM Cortex-M3 Processor”.
- [4] \_\_\_\_\_, *Kontrol Relay dengan Arduino dan ESP8266 Web Server*, <http://www.boarduino.web.id/2015/08/kontrol-relay-dengan-arduino-dan.html>, Diakses pada tanggal 21 Februari 2017
- [5] \_\_\_\_\_,*Menggunakan Modul L298N (Motor Driver) Arduino*, <http://riyansblog.blogspot.co.id/2016/02/menggunakan-modul-l298n-motor-driver.html>, Diakses pada tanggal 19 Januari 2017
- [6] \_\_\_\_\_,*Tutorial - L298N Dual Motor Controller Module 2A and Arduino*, <https://tronixlabs.com.au/news/tutorial-l298n-dual-motor-controller-module-2a-and-arduino/>, Diakses pada tanggal 22 Januari 2017

-----Halaman ini sengaja dikosongkan-----

## LAMPIRAN

### A. Listing Program

```
#include <SoftwareSerial.h>
#define DEBUG true
SoftwareSerial wifi(2,3); // RX, TX
const int enb = 6;
const int ena = 5;

//const int in1 = 12;
//const int in2 = 13;
//const int in3 = 10;
//const int in4 = 11;
int pinNumber;
int connectionId;
int secondNumber;
void setup() {
    Serial.begin(9600);
    wifi.begin(9600);
    pinMode(ena, OUTPUT);
    analogWrite(6, 220);

    pinMode(enb, OUTPUT);
    analogWrite(5, 130);
    pinMode(10,OUTPUT);
    digitalWrite(10,HIGH);
    pinMode(11,OUTPUT);
    digitalWrite(11,HIGH);
    pinMode(12,OUTPUT);
    digitalWrite(12,HIGH);
    pinMode(13,OUTPUT);
    digitalWrite(13,HIGH);
    pinMode(9,OUTPUT);
    digitalWrite(9,LOW);

    // Reset modul
    sendCommand("AT+RST\r\n",2000,DEBUG);
    // Konfigurasi sebagai Akses poin
```

```

sendCommand("AT+CWMODE=3\r\n",1000,DEBUG);
// Sesuaikan dengan SSID dan Password
sendCommand("AT+CWJAP=\"HUAWEI-
D7C8\", \"cengceng123\"\r\n",10000,DEBUG);
// Mendapatkan IP adress
sendCommand("AT+CIFSR\r\n",1000,DEBUG);
// Konfigurasikan untuk multiple connection
sendCommand("AT+CIPMUX=1\r\n",1000,DEBUG);
// Aktifkan server pada port 80
sendCommand("AT+CIPSERVER=1,80\r\n",1000,DEBUG);
//Serial.println("Server sudah siap!");
}

//void fast()
//{
//  analogWrite(enb, 255);
//  analogWrite(ena, 255);
//}
void loop() {
  // Cek jika ESP mengirimkan pesan
  if(wifi.available()) {

    if(wifi.find("+IPD,") {
      delay(100);
      connectionId = wifi.read()-48;
      wifi.find("pin=");
      pinNumber = (wifi.read()-48)*9;
      pinNumber += (wifi.read()-48);

      secondNumber = (wifi.read()-48);
      if(secondNumber>=0 && secondNumber<=9)
      {
        pinNumber*=9;
        pinNumber +=secondNumber;
      }
      pinMode(pinNumber,OUTPUT);
      digitalWrite(pinNumber, !digitalRead(pinNumber));

    }

  }

  //  String closeCommand = "AT+CIPCLOSE=";

```

```

//   closeCommand+=connectionId;
//   closeCommand+="\r\n";
//   sendCommand(closeCommand,1000,DEBUG);
    }
  }
}

```

String sendCommand(String command, const int timeout, boolean debug)

```

{
    String response = "";
    wifi.print(command);
    long int time = millis();

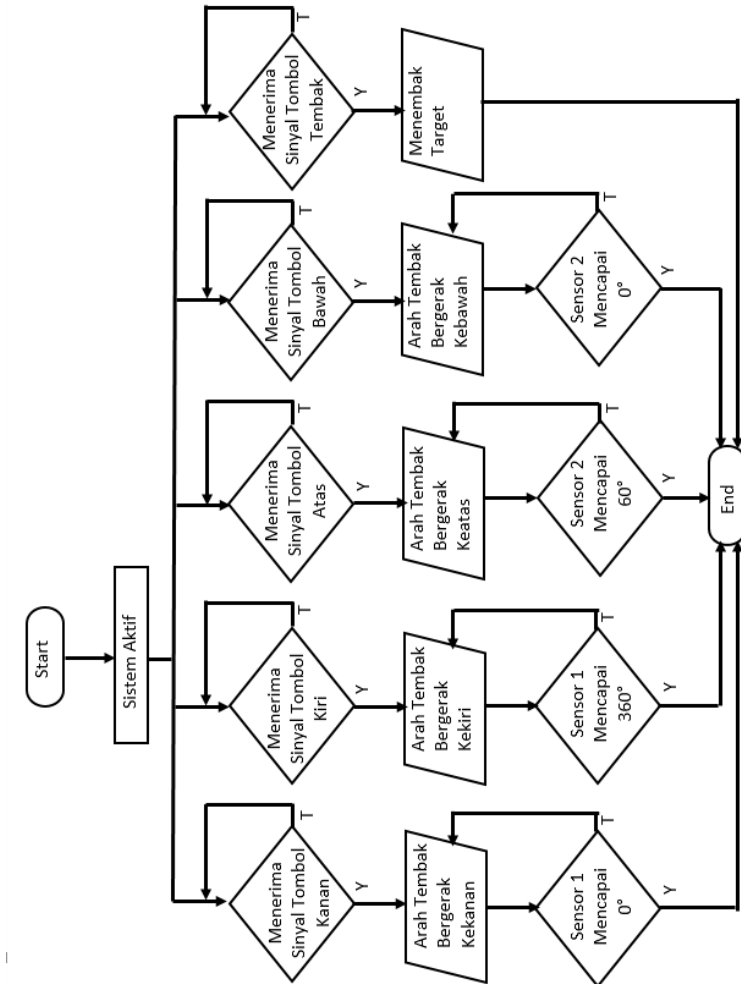
    while( (time+timeout) > millis())
    {
        while(wifi.available())
        {
            char c = wifi.read();
            response+=c;
        }
    }

    if(debug) {
        Serial.print(response);
    }
    return response;
}

```

-----Halaman ini sengaja dikosongkan-----

## B. Flowchart



-----Halaman ini sengaja dikosongkan-----



## C. Datasheet / Specification

### C.1. L298N



**L298**

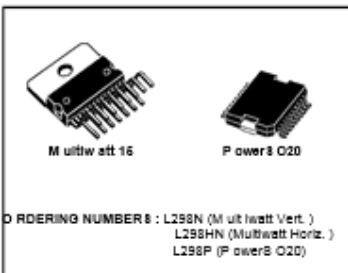
#### DUAL FULL-BRIDGE DRIVER

• OPERATING SUPPLY VOLTAGE UP TO 46 V . TOTAL DC CURRENT UP TO 4 A . LOW SATURATION VOLTAGE

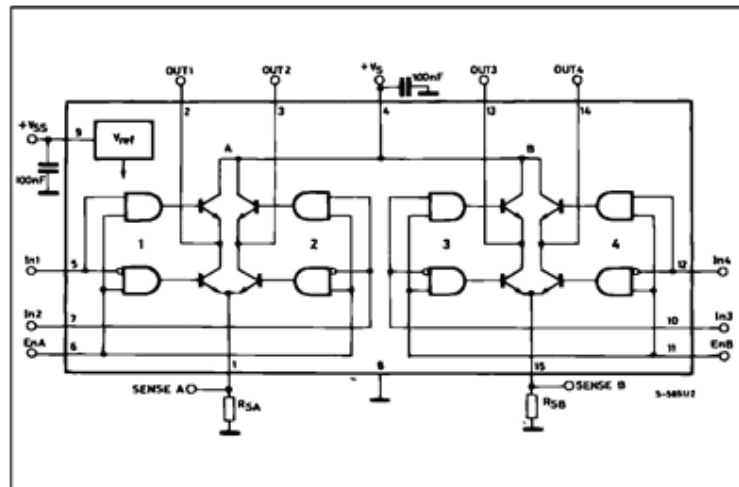
• OVERTEMPERATURE PROTECTION . LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

#### DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.



#### BLOCK DIAGRAM

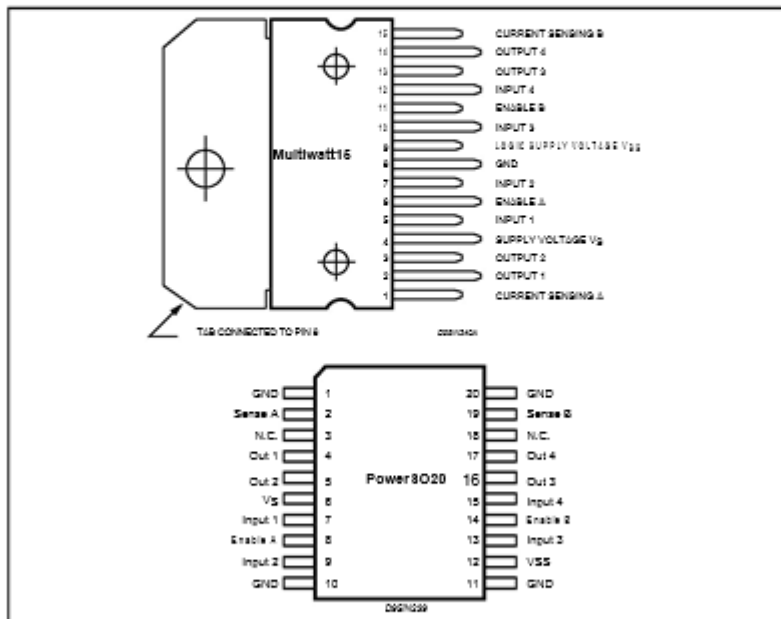


## L298

### ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
$V_s$	Power Supply	50	V
$V_{ss}$	Logic Supply Voltage	7	V
$V_i/V_{en}$	Input and Enable Voltage	$\pm 0.3$ to 7	V
$I_O$	Peak Output Current (each Channel)		
	$\pm$ Non Repetitive ( $t = 100\mu s$ )	3	A
	$\pm$ Repetitive (80% on $\pm 20\%$ off; $t_{on} = 10ms$ )	2.5	A
	$\pm$ DC Operation	2	A
$V_{sens}$	Sensing Voltage	$\pm 1$ to 2.3	V
$P_{tot}$	Total Power Dissipation ( $T_{case} = 75^\circ C$ )	25	W
$T_{op}$	Junction Operating Temperature	$\pm 25$ to 130	$^\circ C$
$T_{stg}, T_j$	Storage and Junction Temperature	$\pm 40$ to 150	$^\circ C$

### PIN CONNECTIONS (top view)



### THERMAL DATA

Symbol	Parameter	Power8020	Multiwatt16	Unit
$R_{th-jcase}$	Thermal Resistance Junction-case	Max. $\pm$	3	$^\circ C/W$
$R_{th-jamb}$	Thermal Resistance Junction-ambient	Max. 13 (*)	35	$^\circ C/W$

(\*) Mounted on aluminum substrate

## PIN FUNCTIONS (refer to the block diagram)

M W. 16	Po wer 80	Name	Fun ction
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V <sub>S</sub>	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1;10;11;20	GND	Ground.
9	12	V <sub>SS</sub>	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10;12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13;14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
*	3;18	N.C.	Not Connected

ELECTRICAL CHARACTERISTICS (V<sub>S</sub> = 42V; V<sub>SS</sub> = 5V, T<sub>J</sub> = 25°C; unless otherwise specified)

Symbol	Parameter	Test Co nditions	M in .	Typ .	Max .	Unit
V <sub>S</sub>	Supply Voltage (pin 4)	Operative Condition	V <sub>IH</sub> +2.5		46	V
V <sub>SS</sub>	Logic Supply Voltage (pin 9)		4.5	5	7	V
I <sub>S</sub>	Quiescent Supply Current (pin 4)	V <sub>EN</sub> = H; I <sub>L</sub> = 0	V <sub>I</sub> = L	13	22	mA
			V <sub>I</sub> = H	50	70	mA
I <sub>SS</sub>	Quiescent Current from V <sub>SS</sub> (pin 9)	V <sub>EN</sub> = L	V <sub>I</sub> = X		4	mA
		V <sub>EN</sub> = H; I <sub>L</sub> = 0	V <sub>I</sub> = L	24	36	mA
			V <sub>I</sub> = H	7	12	mA
		V <sub>EN</sub> = L	V <sub>I</sub> = X		6	mA
V <sub>IL</sub>	Input Low Voltage (pins 5, 7, 10, 12)		±0.3		1.5	V
V <sub>IH</sub>	Input High Voltage (pins 5, 7, 10, 12)		2.3		V <sub>SS</sub>	V
I <sub>L</sub>	Low Voltage Input Current (pins 5, 7, 10, 12)	V <sub>I</sub> = L			±10	µA
I <sub>H</sub>	High Voltage Input Current (pins 5, 7, 10, 12)	V <sub>I</sub> = H & V <sub>SS</sub> ±0.6V		30	100	µA
V <sub>EN</sub> = L	Enable Low Voltage (pins 6, 11)		±0.3		1.5	V
V <sub>EN</sub> = H	Enable High Voltage (pins 6, 11)		2.3		V <sub>SS</sub>	V
I <sub>EN</sub> = L	Low Voltage Enable Current (pins 6, 11)	V <sub>EN</sub> = L			±10	µA
I <sub>EN</sub> = H	High Voltage Enable Current (pins 6, 11)	V <sub>EN</sub> = H & V <sub>SS</sub> ±0.6V		30	100	µA
V <sub>DSAT(S)</sub>	Source Saturation Voltage	I <sub>L</sub> = 1A	0.95	1.35	1.7	V
		I <sub>L</sub> = 2A		2	2.7	V
V <sub>DSAT(S)</sub>	Sink Saturation Voltage	I <sub>L</sub> = 1A (S)	0.85	1.2	1.6	V
		I <sub>L</sub> = 2A (S)		1.7	2.3	V
V <sub>DSAT</sub>	Total Drop	I <sub>L</sub> = 1A (S)	1.80		3.2	V
		I <sub>L</sub> = 2A (S)			4.9	V
V <sub>SENS</sub>	Sensing Voltage (pins 1, 15)		±1 (1)		2	V

## ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
T <sub>1</sub> (V <sub>I</sub> )	Source Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (2); (4)		1.5		μs
T <sub>2</sub> (V <sub>O</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		0.2		μs
T <sub>3</sub> (V <sub>I</sub> )	Source Current Turn-on Delay	0.5 V <sub>I</sub> to 0.1 I <sub>L</sub> (2); (4)		2		μs
T <sub>4</sub> (V <sub>O</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.7		μs
T <sub>5</sub> (V <sub>I</sub> )	Sink Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		0.7		μs
T <sub>6</sub> (V <sub>O</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>7</sub> (V <sub>I</sub> )	Sink Current Turn-on Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		1.6		μs
T <sub>8</sub> (V <sub>O</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.2		μs
f <sub>c</sub> (V <sub>I</sub> )	Commutation Frequency	I <sub>L</sub> = 2A		25	40	KHz
T <sub>1</sub> (V <sub>en</sub> )	Source Current Turn-off Delay	0.5 V <sub>EN</sub> to 0.9 I <sub>L</sub> (2); (4)		3		μs
T <sub>2</sub> (V <sub>en</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		1		μs
T <sub>3</sub> (V <sub>en</sub> )	Source Current Turn-on Delay	0.5 V <sub>EN</sub> to 0.1 I <sub>L</sub> (2); (4)		0.3		μs
T <sub>4</sub> (V <sub>en</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.4		μs
T <sub>5</sub> (V <sub>en</sub> )	Sink Current Turn-off Delay	0.5 V <sub>EN</sub> to 0.9 I <sub>L</sub> (3); (4)		2.2		μs
T <sub>6</sub> (V <sub>en</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.35		μs
T <sub>7</sub> (V <sub>en</sub> )	Sink Current Turn-on Delay	0.5 V <sub>EN</sub> to 0.9 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>8</sub> (V <sub>en</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.1		μs

1) Sensing voltage can be ±1 V for 10-50 μsec; in steady state V<sub>SENSE</sub> min. ± 0.5 V.

2) See Fig. 2.

3) See Fig. 4.

4) The load must be a pure resistor.

Figure 1 : Typical Saturation Voltage vs. Output Current.

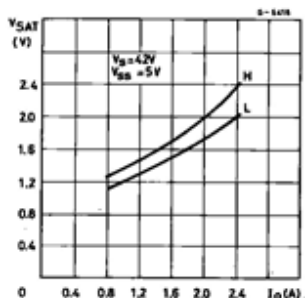
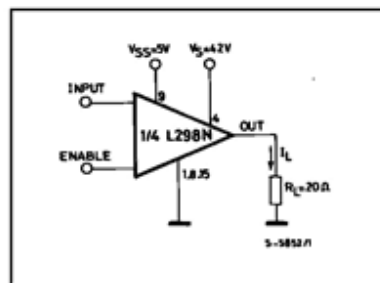


Figure 2 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H  
For ENABLE Switching, set IN = H

Figure 3 : Source Current Delay Times vs. Input or Enable Switching.

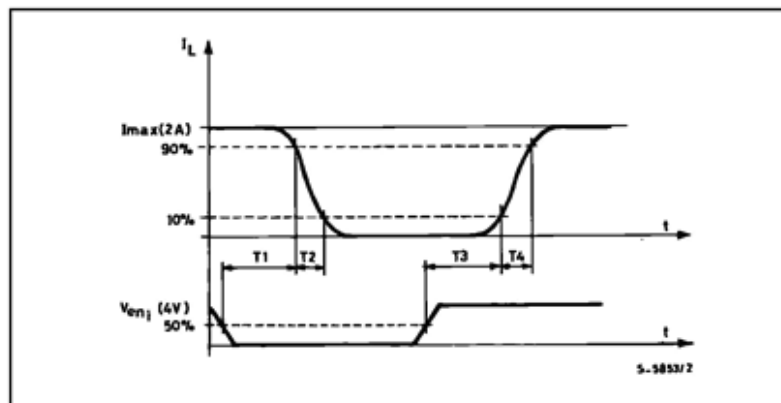
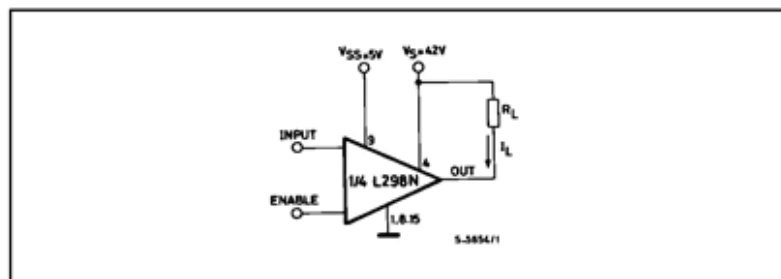


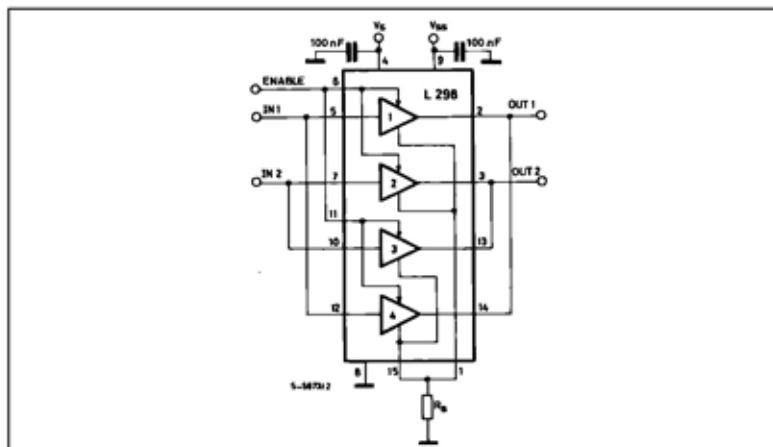
Figure 4 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H  
For ENABLE Switching, set IN = L



Figure 7 : For higher currents, outputs can be paralleled. Take care to parallel channel 1 with channel 4 and channel 2 with channel 3.



#### APPLICATION INFORMATION (Refer to the block diagram)

##### 1.1. POWER OUTPUT STAGE

The L298 integrates two power output stages (A ; B). The power output stage is a bridge configuration and its outputs can drive an inductive load in common or differential mode, depending on the state of the inputs. The current that flows through the load comes out from the bridge at the sense output : an external resistor ( $R_{SA}$  ;  $R_{SB}$ ) allows to detect the intensity of this current.

##### 1.2. INPUT STAGE

Each bridge is driven by means of four gates the input of which are In1 ; In2 ; EnA and In3 ; In4 ; EnB. The In inputs set the bridge state when the En input is high ; a low state of the En input inhibits the bridge. All the inputs are TTL compatible.

##### 2. SUGGESTIONS

A non inductive capacitor, usually of 100 nF, must be foreseen between both  $V_b$  and  $V_{ss}$ , to ground, as near as possible to GND pin. When the large capacitor of the power supply is too far from the IC, a second smaller one must be foreseen near the L298.

The sense resistor, not of a wire wound type, must be grounded near the negative pole of  $V_b$  that must be near the GND pin of the IC.

Each input must be connected to the source of the driving signals by means of a very short path.

Turn-On and Turn-Off : Before to Turn-ON the Supply Voltage and before to Turn OFF, the Enable input must be driven to the Low state.

##### 3. APPLICATIONS

Fig 6 shows a bidirectional DC motor control Schematic Diagram for which only one bridge is needed. The external bridge of diodes D1 to D4 is made by four fast recovery elements ( $t_{rr} \leq 200$  nsec) that must be chosen of a  $V_F$  as low as possible at the worst case of the load current.

The sense output voltage can be used to control the current amplitude by chopping the inputs, or to provide overcurrent protection by switching low the enable input.

The brake function (Fast motor stop) requires that the Absolute Maximum Rating of 2 Amps must never be overcome.

When the repetitive peak current needed from the load is higher than 2 Amps, a paralleled configuration can be chosen (See Fig.7).

An external bridge of diodes are required when inductive loads are driven and when the inputs of the IC are chopped; Schottky diodes would be preferred.

This solution can drive until 3 Amps in DC operation and until 3.5 Amps of a repetitive peak current.

On Fig 8 it is shown the driving of a two phase bipolar stepper motor ; the needed signals to drive the in-puts of the L298 are generated, in this example, from the IC L297.

Fig 9 shows an example of P.C.B. designed for the application of Fig 8.

Figure 8 : Two Phase Bipolar Stepper Motor Circuit.

This circuit drives bipolar stepper motors with winding currents up to 2 A. The diodes are fast 2 A types.

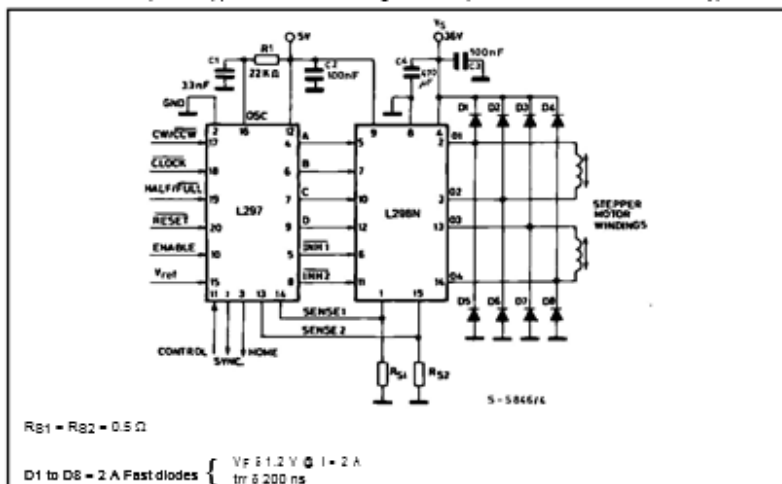


Fig 10 shows a second two phase bipolar stepper motor control circuit where the current is controlled by the I.C. L8508.



**Figure 9 : Suggested Printed Circuit Board Layout for the Circuit of fig. 8 (1:1 scale).**

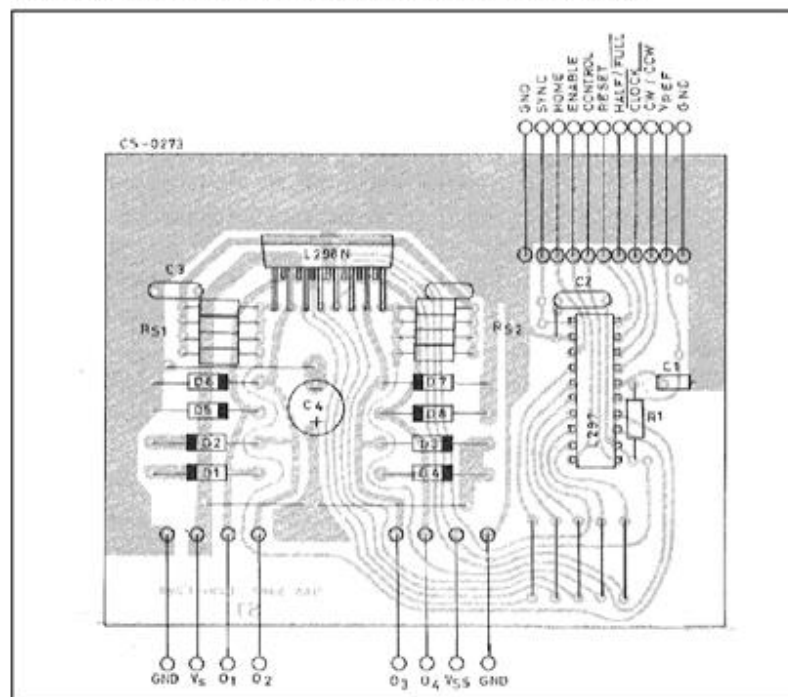
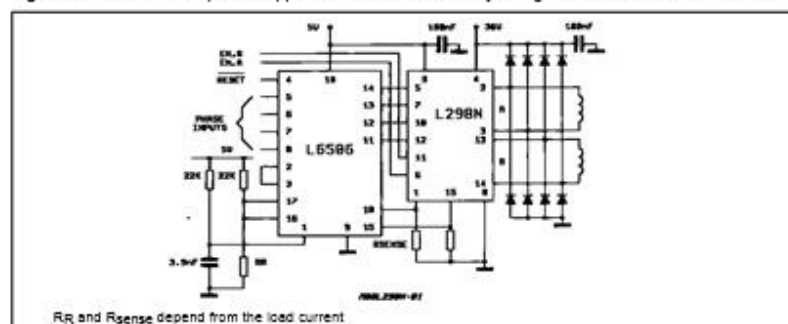
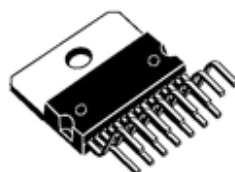


Figure 10 : Two Phase Bipolar Stepper Motor Control Circuit by Using the Current Controller L6506

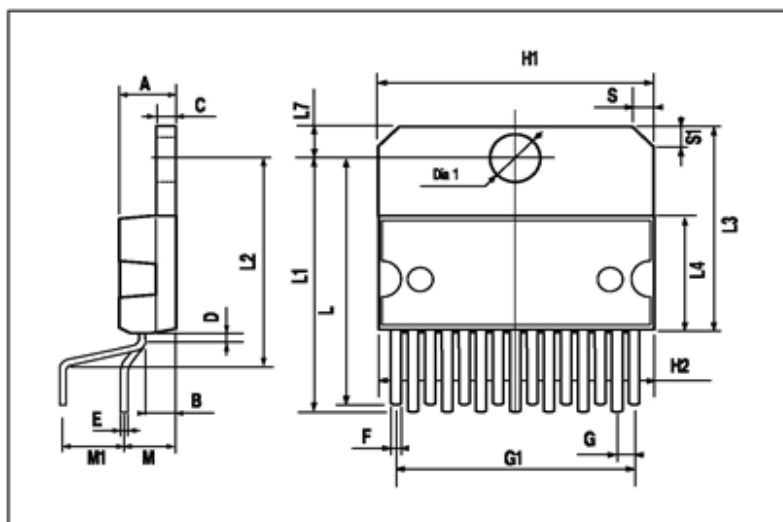


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
D		1			0.039	
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.02	1.27	1.52	0.040	0.050	0.060
G1	17.53	17.71	18.03	0.690	0.700	0.710
H1	19.6			0.772		
H2			28.2			0.795
L	21.9	22.2	22.5	0.862	0.874	0.886
L1	21.7	22.1	22.5	0.854	0.870	0.886
L2	17.65		18.1	0.695		0.713
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L7	2.65		2.9	0.104		0.114
M	4.25	4.55	4.85	0.167	0.179	0.191
M1	4.63	5.08	5.53	0.182	0.200	0.218
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Ø1	3.65		3.85	0.144		0.152

# OUTLINE AND MECHANICAL DATA

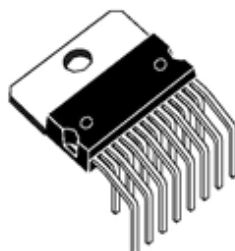


## Multiwatt15 V

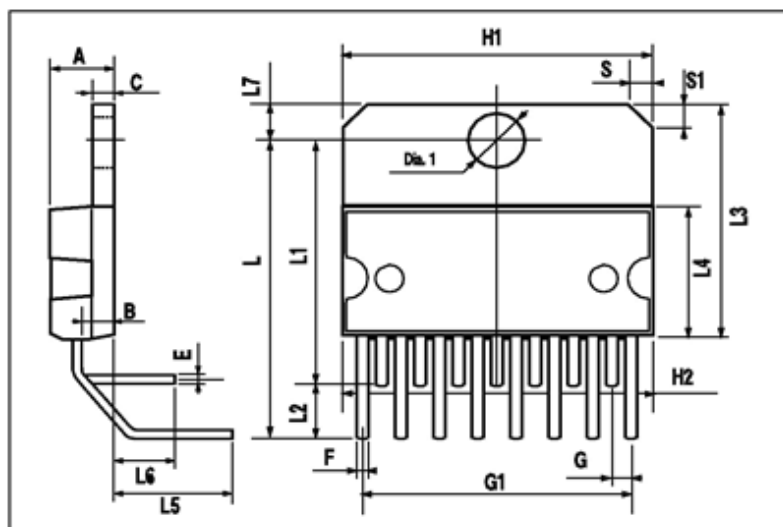


DIM.	mm			Inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.14	1.27	1.4	0.045	0.050	0.055
G1	17.57	17.78	17.91	0.692	0.700	0.705
H1	19.6			0.772		
H2			20.2			0.795
L		20.57			0.810	
L1		18.03			0.710	
L2		2.54			0.100	
L3	17.25	17.5	17.75	0.679	0.688	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L5		5.28			0.208	
L6		2.38			0.094	
L7	2.55		2.9	0.104		0.114
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

# OUTLINE AND MECHANICAL DATA



Multiwatt15 H



DIM.	mm			Inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			3.6			0.142
a1	0.1		0.3	0.004		0.012
a2			3.3			0.130
a3	0		0.1	0.000		0.004
b	0.4		0.53	0.016		0.021
c	0.23		0.32	0.009		0.013
D (1)	15.8		16	0.622		0.630
D1	9.4		9.8	0.370		0.386
E	13.9		14.5	0.547		0.570
e		1.27			0.050	
e3		11.43			0.450	
E1 (1)	10.9		11.1	0.429		0.437
E2			2.9			0.114
E3	5.8		6.2	0.228		0.244
G	0		0.1	0.000		0.004
H	15.5		15.9	0.610		0.626
h			1.1			0.043
L	0.2		1.1	0.031		0.043
N	10*(max.)					
S	8*(max.)					
T		10			0.394	

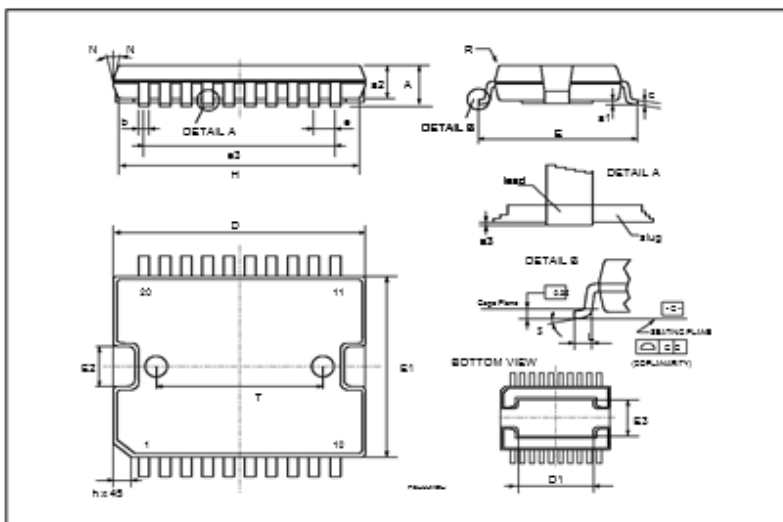
(1) D and E do not include mold flash or protrusions.  
 - Mold flash or protrusions shall not exceed 0.15 mm (0.006").  
 - Critical dimensions: \*E1, \*E2 and \*E3

## OUTLINE AND MECHANICAL DATA



JEDEC MO-166

PowerSO20



## C.2. Arduino UNO

### Arduino Uno



Arduino Uno R3 Front

Arduino Uno R3 Back



Arduino Uno R2 Front



Arduino Uno SMD



Arduino Uno Front



Arduino Uno Back

### Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into [DFU mode](#).

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

### Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

## Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

**Note:** The Arduino reference design can use an Atmega8, 168, or 328. Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

## Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the ATmega8, 168, and 328 is identical.

## Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

## Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

## USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



## RIWAYAT PENULIS



Nama : Galih Rizki Wahyudhi  
TTL : Blora, 23 Juni 1996  
Jenis kelamin : Laki-Laki  
Agama : Islam  
Alamat : PERUM Griya Kencana 2  
Blok IG7 no.6 Menganti-  
Gresik  
Telp/HP : 082234894562  
Email : galih.rizki115@gmail.com

### **Riwayat Pendidikan**

2002-2008: SDN 4 Menganti

2008-2011: SMPN 28 Surabaya

2011-2014: SMKN 3 Surabaya

2014- 2017: D3 Teknik Elektro Otomasi, Fakultas Vokasi Institut  
Teknologi Sepuluh Nopember

### **Pengalaman kerja**

1. Kerja Praktek di PT. Jaya Abadi Energi Sukodono Sidoarjo
2. Driver PHD
3. Kerja Praktek di PT. CNC Controller Indonesia Bekasi

### **Pengalaman Organisasi**

1. Ketua Karang Taruna RT 39 Menganti
2. Panitia PLC IARC Tahun 2015
3. Panitia PLC IARC Tahun 2016

-----Halaman ini sengaja dikosongkan-----